

**RECEIVED
CENTRAL FAX CENTER**

Attorney Docket No.: 40101/02301 (2000.055)

IN THE CLAIMS:**JAN 18 2008**

1. (Currently amended) A system, comprising a processor adapted to execute a software package, ~~the~~ the software package, comprising:

~~a receiving module determining a format of each of a plurality of original files;~~

and

a converter module applying a converter function corresponding to ~~the a~~ a file format of each of ~~the a~~ a plurality of original files to create new files in a converted file format, ~~wherein the converter module includes storing~~ an extensible set of converter functions and the converter function is selected from the extensible set, wherein each of the converter functions within the converter module provides instructions necessary to convert each of the plurality of original files to the new files; and

a receiving module determining a format of each of the plurality of original files, the receiving module forwarding each of the files to the converter module based on the determined format for each of the original files, the receiving module being separate from the converter module.

2. (Original) The software package according to claim 1, further comprising:

an application program to access information in the new files, wherein the application program is compatible with the new files and incompatible with the original files.

3. (Original) The software package according to claim 1, wherein the converted file format is a document object model tree.

4. (Original) The software package according to claim 1, wherein the converter function includes a text parser.

5. (Original) The software package according to claim 1, wherein the original files include a configuration file.

Attorney Docket No.: 40101/02301 (2000.055)

6. (Original) The software package according to claim 1, wherein the receiving module determines the format of the original files based on file extensions.

7. (Currently amended) A system, comprising:

an application module to perform functions, the application module uses information contained in a configuration file to perform the functions; ~~and~~

a conversion module applying a converter function to the configuration file to convert the configuration file from a first format incompatible with the application module to a second format compatible with the application module, ~~wherein the conversion module includes~~ storing an extensible set of converter functions and the converter function is selected from the extensible set, wherein each of the converter functions within the conversion module provides instructions necessary to convert each of the plurality of original files to the new files; and

a receiving module determining a format of each of the plurality of original files, the receiving module forwarding each of the files to the conversion module based on the determined format for each of the original files, the receiving module being separate from the conversion module.

8. (Original) The system according to claim 7, wherein the conversion module includes a plurality of converter functions corresponding to a plurality of file formats, the first format being one of the plurality of file formats.

9. (Original) The system according to claim 8, wherein the conversion module includes a receiving element to determine the first format.

10. (Original) The system according to claim 7, wherein the second format is a document object model tree.

11. (Original) The system according to claim 7, wherein the first format is an extensible markup language.

Attorney Docket No.: 40101/02301 (2000.055)

12. (Currently amended) A method of converting an original file from a plurality of formats, comprising the steps of:

reading a format of the original file;

determining, by a receiving module, a format of each of the plurality of original files;

forwarding, by a receiving module, the original file to a converter module based on the determined format for each of the original files, the receiving module being separate from the converter module;

applying to the original file, by a converter module, a converter function corresponding to the format of the original file, wherein the converter function is one of an extensible plurality of converter functions stored within the converter module, wherein each of the converter functions within the converter module provides instructions necessary to convert each of the plurality of original files to the new files; and

saving the original file in a new file which is in a converted file format created by the application of the corresponding converter function.

13. (Original) The method of claim 12, further comprising the step of:

outputting the new file to an application program which uses information in the new file to configure the application program, wherein the application program is compatible with the new file and incompatible with the original file.

14. (Original) The method of claim 12, wherein the converted file format is a document object model tree.

15. (Original) The method of claim 12 wherein the plurality of formats of the original files include an extensible markup language.

16. (Original) The method of claim 12, wherein the new files are saved in one of random access memory and permanent memory.